DESIGN OF A GROUND-LAUNCHED BALLISTIC MISSILE INTERCEPTOR USING A GENETIC ALGORITHM

Murray B. Anderson[+], Sverdrup Technology Inc., TEAS Group
John E. Burkhalter[*] and Rhonald M. Jenkins[↗], Auburn University

Abstract

For this preliminary design study an apportioned pareto genetic algorithm, unique to the software package IMPROVE©[1], was used to manipulate a solid rocket design code, an aerodynamic design code, and a three-loop autopilot to produce interceptor designs capable of accurately engaging a high-speed/high-altitude target. Twenty-nine design variables were required to define the optimization problem, and four primary goals were established to access the performance of the interceptor designs. Design goals included 1) minimize miss distance, 2) minimize intercept time, 3) minimize takeoff weight, and 4) minimize maximum G-loading. In 50 generations the genetic algorithm was able to develop two basic types of external aerodynamic designs that performed nearly the same, with miss distances less than 1.0 foot. The solid rocket motors that propelled these external shapes shared common characteristics such as a large initial burning area and a large combustion chamber volume. Examination of the intercept trajectories shows that standard proportional navigation guidance works adequately. The three-loop autopilot performs well even for high altitude engagements, and the analytic gain determination makes the autopilot straightforward to implement.

Introduction

With the addition of guidance, an autopilot, and an airframe with movable control surfaces, basic rocketry expands into a more lethal and much more precise means of waging war. Rather than increasing the size of the warhead being delivered (to make-up for a loss in delivery accuracy), modern weaponeering has tended to use a small warhead coupled with an accurate control system. For ground launched systems, ideas such as "smart rocks" and "brilliant pebbles" that sprang from early Strategic Defense Initiative (SDI) research were based on the belief that the energy delivered by a small fast moving projectile without an explosive could be as lethal as a less accurate system with an explosive warhead. Similarly for air-launched ground attack weapons, Vietnam proved that delivery accuracy was paramount to defeating tough targets, and the Air Force devoted billions of dollars to the development of precise warhead delivery systems. Certainly the Persian Gulf War showed the benefit of the highly accurate weapon systems the Air Force developed. In more human terms, increased delivery accuracy means less collateral damage to non-military facilities and less civilian casualties. While it is difficult to argue with the success of the air-launched ground attack weapons that have been demonstrated in combat, these systems, and air-to-air missile systems, share a common design philosophy with the less successful ground launch interceptors. When a system has a guidance system and autopilot, there is a tendency to compensate for less than stellar aerodynamic designs by shifting more and more of the delivery problems over to the autopilot. As a result, autopilots are typically very good and very robust, but the airframe and aerodynamics of the overall system are almost an afterthought. Overall system performance and system capability, therefore, suffers because of the over-reliance on the autopilot to compensate for weaknesses in the aerodynamic performance of the weapon system. The goal of this research is to let an artifiicial intelligence tool, a genetic algorithm, design the aerodynamic shape while at the same time designing the propulsion system and key autopilot variables. This all-at-once approach to missile design is intended to provide a system capable of producing good aerodynamic shapes in addition to the good performance expected from an autopilot.

Previous Work

Several applications of genetic algorithms to autopilot and control systems optimization are worth noting. Norris and Crossley[2] recently used a genetic algorithm to find gains to control a standard pedagogical two-disk torsional spring system. The approach used a very simple two-loop proportional-integral control system

---

[+] Deputy Director, Development Test Department, Senior Member, AIAA
[*] Professor, Auburn University, Associate Fellow, AIAA
[↗] Associate Professor, Auburn University, Senior Member, AIAA

with velocity feedback. The two loop controller had three gains that needed to be determined as a function of variable spring stiffness. The objective functions were defined such that good gain values produce little error between the commanded and achieved disk rotation angles. Since two separate disks were being commanded in twist, a pareto genetic algorithm was used to try to minimize the rotational errors in both disks simultaneously. At the conclusion of 80 generations, the resulting family (i.e. population of 80 members) of three controller gains were hybrid performers that would work reasonably well for both disks. Two points worth noting from this study was that the crossover probability was rather low (50 percent) and the population size was three times the number of bits required to represent the three gains (12 bits per gain, 36 bits total).

Another interesting genetic algorithm controller application was recently presented by McGookin[3] to control the steering of oil tankers to multiple waypoints in a narrow channel. In McGookin's work the control system consisted of a two-loop autopilot and a sliding mode controller (SMC). The four parameters being optimized consisted of the 1st and 2nd heading loop poles (frequency plane), a heading switch gain, and a so-called heading boundary layer thickness. The goal (e.g. objective function) of the study was to find values for these four parameters such that the oil tanker passed within an acceptable distance of each waypoint while minimizing rudder movement. Minimizing rudder movement saves fuel and time. The results shown indicate that the genetic algorithm found excellent values of the four "gains" in 100 generations. Rudder movement was minimal and each waypoint was reached with very little error. In terms of the genetic algorithm operation in this study, it is interesting that McGookin used a 5 percent mutation rate, which is at least an order of magnitude higher than conventional values of this parameter. Since there is no agreement among genetic algorithm researchers about value ranges for crossover and mutation, it is interesting to note what values other researchers use for their applications.

Another study by Martin[4] addressed the issue of autopilot gain scheduling by using genetic algorithms to replace the ad hoc design process typical in linear gain scheduling with a genetically-fit hyperplane-surface strategy. The genetic algorithm was basically used to optimally design the gain schedule. Of course a gain scheduling approach that might work for one scenario could be inadequate without adaptation. Adaptation strategies have been pursued by Karr and Harper[5] using genetic algorithms to augment fuzzy logic controllers. Coupling genetic algorithms with neural networks appears to offer improvement to adaptive control that neither approach has independently. The fuzzy controller (neural network) uses a "rule-of-thumb" strategy to control a chemical system, but the chemical system is periodically changed, thereby invalidating some of the rules-of-thumb. As the system changes, a learning algorithm (the genetic algorithm in this case) tests new rules-of-thumb so that the fuzzy controller can continue to control the chemical system. For autonomous systems[6] this type of approach has obvious advantages assuming the genetic algorithm can keep up with the rate of change of the system. Karr[7] later expanded this work to control the rendezvous of two spacecraft. Another excellent work in adaptive fuzzy logic controller design using genetic algorithms was done by Homaifar and McCormick[8] to control a simple electronic cart. The genetic algorithm designed both the rules-of-thumb and the membership functions for the system in an automated process that did not require human input.

Guidance Algorithm

The guidance algorithm used in this study is standard proportional navigation (called ProNav). A ProNav system commands accelerations normal to line of sight between the missile and the target, proportional to the closing velocity (V_c) and the line of site rate ($\dot{\lambda}$). In equation form, this relationship is

$$\eta_c = N' V_c \dot{\lambda} \qquad (1)$$

where N' is the effective navigation ratio or gain. The closing velocity and line of sight rate are typically determined by a Doppler radar and seeker respectively. For this research it is assumed that there is a perfect seeker and a perfect radar system so that the target position and velocity are known exactly. For preliminary design studies these two assumptions are appropriate. However, to make the autopilot performance variables (like damping ratio) more meaningful, the target position/velocity model was run at a slightly slower time step than the autopilot itself. Typical ranges for N' are 3 to 5 (unitless) according to Zarchan[9] for tactical weapon systems, so given this variation the effective navigation ratio is a variable the genetic algorithm should determine.

As the missile enters the endgame maneuvers near the target, the line-of-sight rate will approach infinity as the missile passes (or passes through) the target and therefore the commanded accelerations will also approach infinity. It is common to limit the total acceleration commands (circular total acceleration commands) for flight systems, and for this study the total acceleration was limited to 90 G's. Lateral accelerations greater than 90 G's would likely cause failure of the missile electronic components if not the structure itself.

Autopilot

The autopilot chosen for this study is the so-called three-loop pitch/yaw autopilot. This autopilot design was chosen because of its simplicity, because it is actively

being used in several existing weapon systems, and because it is possible to analytically calculate the proper system gains (for all flight conditions) based on a few specified autopilot performance parameters. These performance parameters are the damping ratio (ζ), time constant (τ), and crossover frequency (ω_{cr}). The system damping ratio governs the sensitivity of the system (i.e. small heading errors should not produce large elevator/rudder deflections) and limits overshooting the commanded acceleration to help protect the structural integrity of the system and the electronics. The time constant is a measure of how fast the system responds to acceleration commands. The crossover frequency (when gain falls below 0.0Db) is a measure of autopilot robustness when higher order dynamics are not modeled. Since the system analyzed here is a 5th order system, the crossover frequency essentially determines how fast the autopilot responds during homing. Higher crossover frequency values mean a faster autopilot, but if the autopilot is too fast it can cause instability in the actuator. A good rule of thumb[9] is that the crossover frequency should not exceed 1/3 of the bandwidth of the actuator. In this study the actuator was assumed to have a natural frequency of 125 rad/sec and a damping of 0.7, so the highest crossover frequency that can be safely used should be around 41.66 rad/sec to maintain stability. This particular instability is the reason why in preliminary design studies, determining a good estimate of the crossover frequency can only really be done with a 5th order system or higher. Evaluation of 3rd order systems (perfect actuator) is usually a first step to analyzing a 5th order system (actuator included), followed by an 11th order system that includes actuator dynamics, IMU response, and structural dynamics[10].

The equations that determine the four autopilot gains (at a single flight condition) from the aerodynamics and autopilot performance parameters are omitted in this discussion. These equations are readily available from multiple sources, however, it should be noted that Nesline[10] provides general equations that work well even for very unstable missiles while Zarchan[9] does not. The basic premise of these equations is that the aerodynamic derivatives (like $C_{m\alpha}$ and $C_{m\delta}$) and expected aerodynamic damping provide a means of determining the linear response that a system is capable of delivering at a given dynamic pressure. If the system response can be estimated, appropriate gain schedules can be developed to achieve the desired autopilot performance levels.

Aerodynamics Code

Washington[11] developed AeroDesign, the aerodynamic prediction methodology used as the basis for this study. AeroDesign was modified to include two axial force considerations that were not part of the original software. First, the fineness ratio of the nose of the missile is compared to a Sears-Haack[12] body and if the nose is not slender enough a drag penalty proportional to the nose bluntness is added to the baseline axial force coefficient. The second axial force coefficient correction was implemented to correct for cases where the rocket nozzle exit diameter actually exceeds the diameter of the body.

AeroDesign was further modified to provide aerodynamic damping derivative estimates and linear aerodynamic coefficient contributions for deflected control surfaces in the pitch and yaw planes in a format and flight condition range compatible with a guided six-degree-of-freedom simulation.

Rocket Performance Code

The solid rocket performance software used in this study is an erosive burning star grain design program that is suitable for preliminary design studies. There are some fundamental assumptions made in the formulation of the software that suitable for rapid evaluation of preliminary design. The major assumptions are:

1. The pressure varies throughout the chamber, however, the pressure is calculated only at the head end (P_1) and at the grain end (P_2). The chamber pressure (P_{CH}) is then defined as the average of these two pressures.
2. The burn rate of the propellant also varies over the entire surface and is subject to erosive burning. The burning rate at any point can be defined as $r = aP_{CH}{}^{n}(1 + k \cdot V)$, where k, a and n are burning rate constants. Since the chamber pressure is calculated only at the head end where V_1 equals 0, and at the aft end of the grain, where V equals V_2, the velocity is averaged to calculate an average erosive burning rate.
3. The grain burns on the edges normal to the centerline of the rocket only (i.e. no end-burning so x_{gl} is constant) at the average burning rate.
4. The flow is isentropic between the aft end of the grain and the throat.
5. The flow obeys the perfect gas law.
6. The chamber pressure varies with time, but is essentially constant during the discharge of a single particle.
7. The flow is one-dimensional and steady.
8. There is no deformation of the propellant due to acceleration, pressure, or viscous forces.
9. The temperature is uniform throughout the grain, but the grain is temperature sensitive (this is a fuel characteristic).

The rocket motor to be designed by the genetic algorithm has certain definable characteristics, such as the strength of the combustion chamber material, which should be known before the design process begins. For this study, the following rocket characteristics were used:

1. Propellant is ammonium perchlorate (80%).
2. Initial temperature of the propellant is 20°C.
3. The design chamber pressure (i.e. maximum chamber pressure for case structural design) is 3000 psi. Chamber case thickness is determined by using a factor of safety of 1.5 given this maximum chamber pressure.
4. The allowable stress in the case is 195,000 psi.
5. The factor of safety is 1.5.
6. The case is made of a steel alloy with a density of 0.28 lbm/in^3.
7. The nozzle is made of an aluminum alloy with a density of 0.19 lbm/in^3.

For ammonium perchlorate the erosive burning rate constant k is 1.0E-4 sec/ft, and the burning rate constants a and n are 0.15 and 0.4 respectively.

Six-Degree-of-Freedom (6-DOF) Simulation

The equations of motion used in this 6-DOF were obtained from Etkin[13] assuming that (1) the missile structure is essentially rigid, (2) there are no rotating masses internal or external to the missile, and (3) there are no significant cross products of inertia. Using Etkin's notation, longitude is measured positive from west to east which is opposite from conventional world maps. Also consistent with Etkin, the standard Eulerian definitions of Φ, Θ, and Ψ were used. To avoid gimbal lock (θ=90 deg) concerns, quaternions were used instead of the standard Euler angles during the integration process.

The conventional body-fixed acceleration and moment equations were modified to include aerodynamic damping terms and contributions due to control surface deflections. For this study a tail-control missile was assumed, but there is no reason why a canard-control system could not also be analyzed. The definitions used for the control surface contributions to the aerodynamic coefficients were as follows:

1) positive elevator commands produce positive normal force contributions and negative pitching moment contributions,
2) positive rudder commands produce positive side force contributions and negative pitching moment contributions.

For a 5th order system, 19 differential equations must be solved simultaneously (15 for the quaternion system of equations and 4 for the elevator and rudder).

Link to GA: GA with Aerodynamics, Propulsion, Six-DOF, Guidance, Autopilot, and Target

Linking the separate software codes to the genetic algorithm was done in a modular fashion so that other modules could be later substituted for the ones used in this study. The genetic algorithm passes all design variables down to the Six-DOF via one subroutine call statement (single line of interface). The Six-DOF then calls the other components, including the mass properties routine, that calculates the component inertias and the center of gravity for the system. For this study the payload was assumed to weigh 50 lbf and the electronic components/actuators weighed an additional 50 lbf.

Variables Governing Design

There are nine variables that govern the solid rocket motor design, fourteen variables that govern the external shape of the vehicle, two variables that control the launch angle (verticality and heading), three variables that define the autopilot performance, and one variable to set the effective navigation ratio or gain. Though the nozzle is shown for the designs presented in this paper, it is merely for visualization purposes. The nozzle actually resides within the total length of the missile. The nozzle exit radius is not, however, free from external aerodynamic considerations since there is a substantial drag penalty that can be incurred if the nozzle exit radius exceeds the body radius. Hopefully the genetic algorithm will learn to design the rocket motor and external shape cooperatively so that good thrust levels are obtained without incurring a drag penalty. Basically then, all outer body dimensions are controlled by the genetic algorithm, from the nose length to the nozzle exit radius.

Table 1 formally defines each design variable and Table 2 shows the minimum, maximum, and resolution that is desired for each variable. The maximum, minimum, and resolution dictate the size of the optimization space. The genetic algorithm requires parameter bounds and resolutions only, and from this table it is obvious that a very broad range of designs is possible. In fact, 2^{175} possible designs exist. The size of this problem is tremendous, especially when the number of atoms in the universe is estimated at 2^{266}.

Table 1. Design Variables for Guided System

Variable Name	Definition (units)
R_{bi}	Grain outer radius, also the rocket motor case inner radius (inches)
R_p	Outer star radius (inches)
R_i	Inner star radius (inches)
x_{gl}	Grain Length (inches)
N_{st}	Number of star points
f_r	Fillet radius (inches)
ε	Angular fraction (rad)
D^*	Diameter of the throat (inches)
R_{exp}	Nozzle expansion ratio (area of throat/area of exit)
Nose	1 – Ogive, 2 - Cone
L_{nose}	Nose Length (inches)
L_{tot}	Total Body Length excluding nozzle (inches)
R_{body}	Body radius (inches)
b_w	Exposed semi-span of wing (inches)
C_{rw}	Wing Root chord (inches)
λ_{tew}	Wing Trailing Edge Sweep Angle (deg)
TR_w	Wing Taper ratio (C_t/C_r)
X_{lew}	Distance from nose tip to wing leading edge (inches)
b_t	Exposed semi-span of tail (inches)
C_{rt}	Tail Root chord (inches)
λ_{tet}	Tail Trailing Edge Sweep Angle (deg)
TR_t	Tail Taper ratio (C_t/C_r)
X_{let}	Distance from nose tip to tail leading edge (inches)
θ	Euler Vertical Launch Angle (θ=90 would be vertical (deg))
Ψ	Euler Launch Heading Angle (Ψ=90 would be East (deg), 0 would be North)
ζ	Damping Ratio
τ	Time Constant
ω_{cr}	Crossover Frequency
N'	Effective Navigation Ratio or Gain

Table 2. Maximum, Minimum, and Resolution of Variables for Guided System

Parameter	Minimum	Maximum	Resolution	Number of Genes
R_{bi}	2.0	10.0	0.02	9
R_p	0.2	9.9	0.02	9
R_i	0.1	9.5	0.02	9
x_{gl}	50.0	200.0	1.0	8
N_{st}	3	10	1	3
f_r	0.05	1.0	0.01	7
ε	0.25	1.0	0.01	7
D^*	0.1	9.0	0.01	10
R_{exp}	1.0	20.0	0.2	7
Nose	0	1	1	1
L_{nose}	20.0	90.0	5.0	4
L_{tot}	50.0	450.0	10.0	6
R_{body}	3.0	20.0	1.0	5
b_w	0.0	80.0	1.0	7
C_{rw}	0.0	80.0	1.0	7
TR_w	0.0	1.0	0.1	4
λ_{tew}	0.0	44.0	2.0	5
X_{lew}	20.0	200.0	5.0	6
b_t	0.0	80.0	1.0	7
C_{rt}	0.0	80.0	1.0	7
TR_t	0.0	1.0	0.1	4
λ_{tet}	0.0	44.0	1.0	5
X_{let}	200.0	400.0	5.0	6
θ	5.0	90.0	1.0	7
ψ	0.0	180.0	1.0	8
ζ	0.2	1.0	0.05	4
τ	0.1	0.9	0.1	3
ω_{cr}	10.0	100.0	5.0	5
N'	2.0	5.0	0.1	5

Mode of Operation of Genetic Algorithm

There are unique algorithms in the IMPROVE© software package that were developed for multi-objective optimization. The primary new algorithm is called an "Apportioned Pareto" algorithm, which allows favoritism among goals. A complete discussion of the implementation of this algorithm, and the other genetic algorithm parameters listed below that control the genetic algorithm, can be found in reference 14.

Mode/Variable	Value
Apportioned Pareto Domination Strength	4.0
Number of Goals	4.0
Elitist	True
Creep Mutation	True
Remove Duplicates	True
Number of Members of the Population	150
Crossover Rate	90%
Mutation Rate	0.2%
Creep Rate	2%

Design Conflict Checking

Some obvious geometrical checks were used to keep the genetic algorithm from expending computational resourses for designs that were not practical. Seven separate checks were made as follows:

1. Outer rocket motor case radius cannot exceed body radius,
2. Rocket motor grain length cannot exceed body length,
3. Tail control surfaces cannot be coincident with or in front of wing,
4. Tail control surfaces cannot overhand the aft end of the missile,
5. Wing cannot overhand nose portion,
6. Based on the specified payload and electronic weights and densities, and the rocket motor size, the total volume of the missile must be able to house these components,
7. Tail control surfaces must be located such that the actuator hinge line (assumed to be at the 50% location of the tail root chord) can be placed at or very near the rocket motor throat. Since the actuators take up a considerable volume, it is logical that they would be placed near the throat of the nozzle.

If any of these conflicts occur, the genetic algorithm is sent back extremely poor performance values in each goal area so that it will learn not to try these designs in the future.

Thermal and Structural Considerations

The 6-DOF software calculates stagnation temperature at each time step based on Mach number and altitude (standard atmosphere is assumed). If the stagnation temperature ever exceeds 2500 degrees Rankine then the system is assumed to fail because of thermal loads.

The structural considerations are manifested in the strength of the wings and tails since these are obvious weak points. Wing and tail loads during flight are used to calculate root bending moments and bending stresses. If the bending stresses ever exceed 185,000 psi (typical for stainless steel) the wing or tail surfaces fail and the flight is terminated at that point. The wing joints are assumed to be rigidly connected to the missile body along the entire root chord. Each actuator-controlled tail surface is assumed to be mounted on a 1.25-inch diameter stainless steel rod. The genetic algorithm must learn to design systems that will not fail either thermally or structurally.

The Target

The target specified for this research is a fast point-mass ground-attack re-entry vehicle like a SCUD missile. The target subroutine defines initial positions, velocities, and accelerations for the target, and performs simple Euler integration on the equations of motion to update the target's position and velocity in time. For this study, the following target parameters were used:

Variable	Value (units)
Initial Downrange Location	120,000 (feet)
Initial Crossrange Location	50,000 (feet)
Initial Altitude	250,000 (feet)
Initial Downrange Velocity	-3000 (ft/sec), toward interceptor
Initial Crossrange Velocity	0 (ft/sec)
Initial Vertical Velocity	4500 (ft/sec), down toward ground
Downrange Acceleration	30 (ft/sec^2), decelerating toward interceptor
Crossrange Acceleration	0 (ft/sec^2)
Vertical Acceleration	-10 (ft/sec^2), decelerating vertically toward interceptor

These parameters simply model the type of trajectory that could be expected by a re-entry vehicle as it slows (due to increased aerodynamic drag) toward impact. Since there is a fairly large crossrange and the vehicle would over-fly the interceptor launch position, the scenario that is being tested is a perimeter defense system.

Goals for Guided Interceptor

The goals were, in order, to 1) minimize miss distance, 2) minimize intercept time, 3) minimize takeoff weight, and 4) minimize the maximum g-loading experienced by the missile. Since an apportionment pareto genetic algorithm was used, goal order is important.

Results

With the design problem and parameters completely defined, the apportioned pareto genetic algorithm was executed until satisfactory missile performance was obtained. Figure 1 shows the convergence history for each goal. As this figure shows it took the genetic algorithm 12 generations (1800 attempts) before it found a design that was capable of even lifting off the ground. Designs that could not produce thrust were given a miss distance error of 1E+5 so that the genetic algorithm would learn that these were very bad designs. Once the genetic algorithm learned how to produce thrust, the miss distance fell to 298,000 feet at generation 12. This large miss distance was due to tail fin failure just after liftoff. Luckily, within two more generations (generation 14) the genetic algorithm found a design that would not fail structurally or thermally. This design flew to within 30 feet of the target. Such a large change in performance is rather unusual, and highlights what

can happen if the right crossover or mutation occurs at the right place at the right time. This is not to say that the best design of generation 12 actually produced the design at generation 14, but that the surviving genes from generation 12 combined with other survivors and produced a 2nd generation descendant with substantially improved performance.

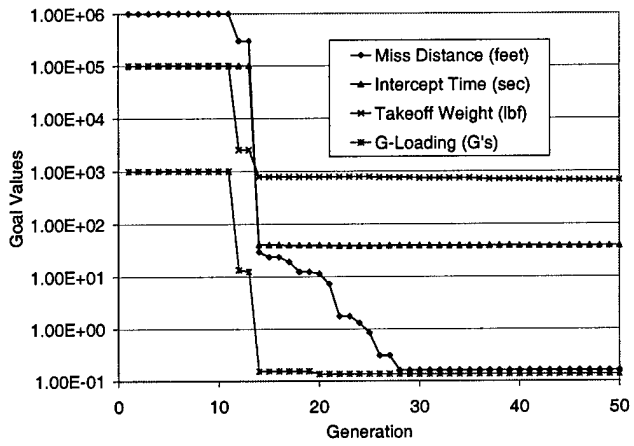

Figure 1. Convergence History for Guided Interceptor Goals

The minimum miss distance continued to improve from generation 14 to generation 28, falling to within 0.16 feet. Given the time step that was used for the last one-second of the engagement, the minimum miss distance could only be 0.1 feet at best, so it is fair to say that the accuracy of the system was near a maximum by generation 28. Prior to the last one-second of flight, the time step was such that the system accuracy could be no better than approximately 5 feet. A larger time step saves considerable computer run time, and for preliminary design efforts saving computer time is important. No further improvement in the minimum miss distance was seen between generation 28 and generation 50.

Though difficult to see in Figure 1 because of the scale, the minimum intercept time fell from 41.6 seconds to 38.5 seconds between generations 14 and 50, and the takeoff weight fell from over 2500 pounds to less than 700 pounds. The convergence figure does not imply that the 700 pound rockets actually hit the targets, nor does it imply that the designs that pulled less than 0.2 G's hit the target, rather, this figure merely shows that within the 150 members of each population the lightest rocket capable of liftoff weighed less than 700 pounds and at least one rocket never pulled more than 0.2 G's during its flight. Analysis of these particular systems showed that the light rockets didn't fly very far and the low-G rockets barely lifted off.

Since there were multiple goals involved in the design process, it is appropriate to examine the history of each goal area through the generations. Figure 2 shows miss distance and intercept time for several generations. It is clear that more and more members of the population maneuver closer and closer to the target as the generations progress. In generation 14, 36 members of the population (36 out of 150) had a miss distance within 100 feet. By generation 50, 124 members of the population reached within 50 feet of the target. By inspection it is also clear from this figure that intercept time falls appreciably between generations 30 and 50 even though the two populations have similar miss distance distributions. Of solutions with a 5 foot miss distance, intercept times varied between 38.8 seconds and 42.3 seconds.

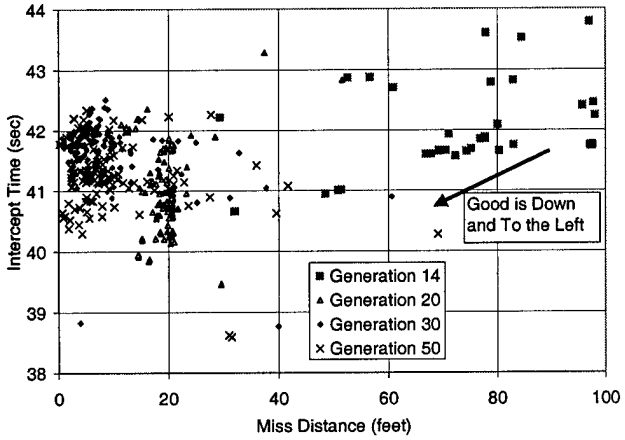

Figure 2. Miss Distance and Intercept Time: Generations 14, 20, 30, and 50

Takeoff weight and miss distance for the same four generations are shown in Figure 3. It is obvious from this figure that the genetic algorithm increased takeoff weight from generations 30 to 50 (likely through the addition of fuel) in order to yield the decreased intercept times shown in Figure 2. This was an expected result, however, it should be realized that continuing to increase fuel mass while decreasing intercept times has a limit when thermal considerations are involved. If the speed of the rocket becomes too high the thermal loads will cause the rocket to fail, and the net result will be large miss distances not decreased intercept times. It is interesting to note that the majority of the designs fall within 175 pounds of a mean weight of roughly 1600 pounds.

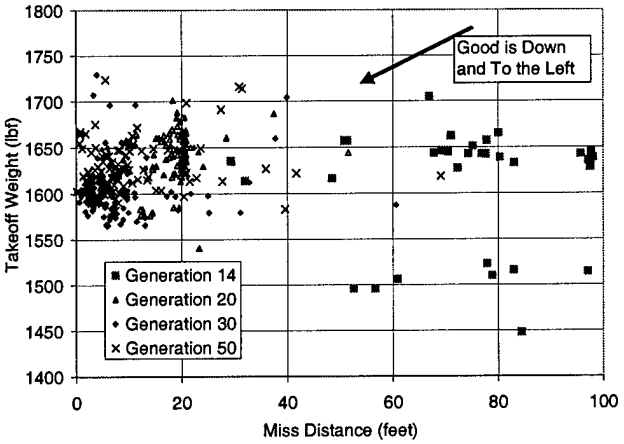

Figure 3. Miss Distance and Takeoff Weight: Generations 14, 20, 30, and 50

Figure 4 shows the maximum G-loading and miss distance for these same four generations. Generations 30 and 50 show that about 40G's is the maximum G-loading that can be expected during the missile flight. This G-loading is certainly within the capability of existing electrical and mechanical systems. From generation 20 to generations 30 and 50 there is a clear trend toward minimizing the G-loading. An average reduction of 5G's occurs between generation 20 and generation 50. There is no clear improvement between generations 30 and 50.

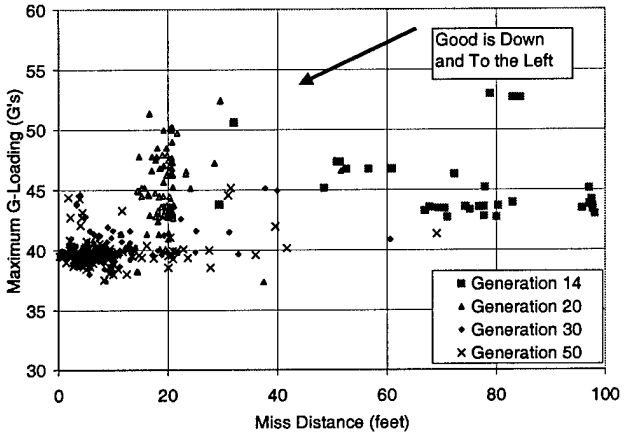

Figure 4. Miss Distance and G-Loading: Generations 14, 20, 30, and 50

Figure 5 shows the prevalent time constants and damping ratios that dominated generation 50. There are clearly not 150 individual points (representing members of the population) on this figure because by generation 50 many members of the population were using exactly the same damping ratios and time constants. High system damping is obviously preferred. This result should be expected since overshooting acceleration commands is not a desirable missile flight characteristic because it wastes energy. The preferred time constants were in the 0.5 to 0.6 second range, which is very reasonable since the target is not conducting evasive maneuvers to escape the interceptor. Missiles that are designed to intercept high-G (9-G's is typical) maneuvering targets have time constants near 0.2 to 0.3 seconds so that they can quickly respond to target evasive tactics.

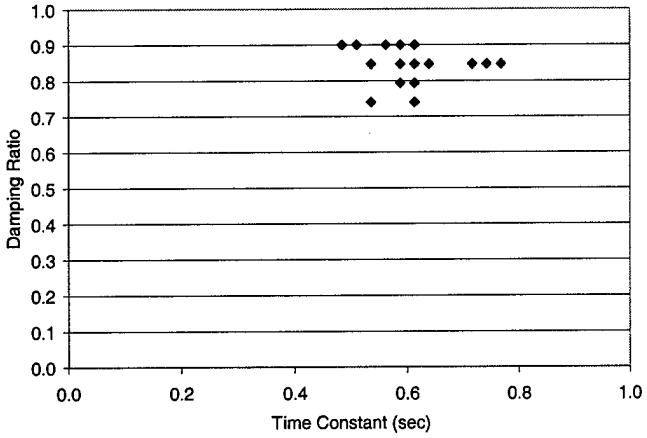

Figure 5. Generation 50 Time Constants and Damping Ratios

Figure 6 shows the proportional navigation gains and crossover frequencies that dominated the population at generation 50. Fairly high navigation gains (4.7-5.0) dominated the population, which means that system quickly tried to minimize heading errors. Low values of the navigation gain, in the 2.0 to 3.5 range, would tend to delay correcting heading errors. For high altitude intercept missions it makes sense to take out heading errors early in the flight rather than waiting until the altitude is such that system responsiveness suffers from the lack of air density (i.e. dynamic pressure). The dominant crossover frequencies were between 20 and 30 rad/sec. This result is not too surprising since the highest value that could safely be used[9] was roughly 41.66 rad/sec, which corresponds to 1/3 of the bandwidth of the actuator used in this study.

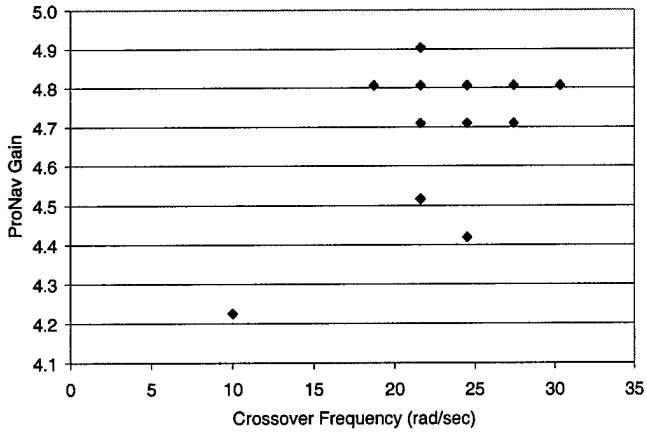

Figure 6. Generation 50 Proportional Navigation Gains and Crossover Frequencies

Before proceeding to look at some of the designs developed by the genetic algorithm, it is interesting to note what initial Euler launch angles were preferred at generation 50. The launch angles were in a fairly tight band, generally between 52 and 56 degrees verticality with a 60 to 63 degree heading. The target initially is located at a 67.38 degree heading angle, which decreases with time because of the negative downrange velocity component, so the genetic algorithm has chosen to lead the target by a few degrees at launch. The target also has an initial elevation angle of 62.52 degrees (decreasing also because it is descending), and the genetic algorithm has chosen to lead the target vertically by a similar magnitude at launch.

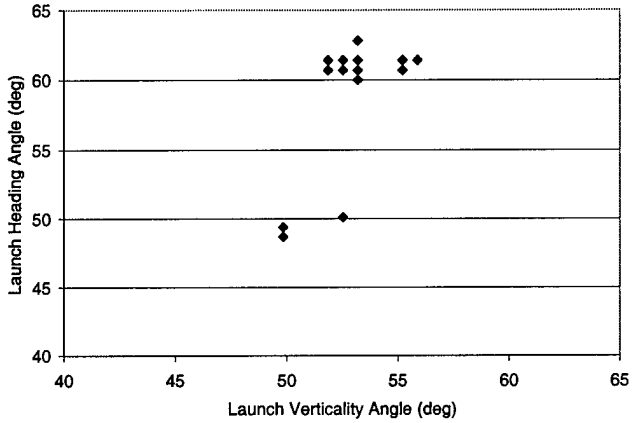

Figure 7. Generation 50 Preferred Euler Launch Angles

Although neither the launch angles nor the autopilot performance parameters show great variation among the members of the population, the actual missile designs produced during the solution process are quite diverse. Figure 8 through Figure 10 show the diversity that exists in a population. The placement of the control surfaces dictates where the throat of the rocket motor is placed (to make room for the actuators within the missile body), so the length of the nozzle expansion region varies. In no case, however, did the actual exit area exceed the diameter of the missile body. Designs yielding nozzle exit radii exceeding the radius of the missile body would produce excess drag, and the genetic algorithm learned to avoid these types of designs. First though, the rocket motor grain designs appear to be very similar, but Figure 8 shows examples of 8, 9, and 10 pointed star grains. A common feature in generation 50 was also present in generation 30 and generation 40, namely large initial burning areas and large combustion chambers.

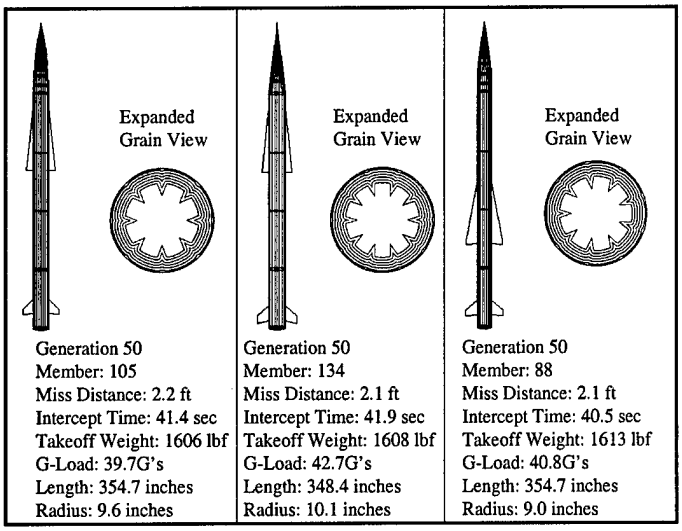

Figure 8. Three Generation 50 Designs with Miss Distances Less Than Three Feet

Nose shapes also continue to vary between ogive's and cone's, although it appears that the ogive nose exists in the more accurate examples. Physical sizes and takeoff weights of the interceptors are very similar, as are intercept times and maximum G-loads. The three best designs (in terms of miss distance only) all had 9-pointed star grains, but the wing sizes and locations still vary significantly. It is also interesting to note that these designs all had nose shapes that were fairly blunt compared to the other designs that were slightly less accurate. These nose shapes were not blunt enough to incur a drag coefficient penalty larger than 0.012 based on a Sears-Haack body, so an examination of the aerodynamic data for these shapes revealed that the net effect of the change in the nose length was to move the center of pressure farther forward very slightly (an average of approximately 1.5 inches) over all flight conditions. This center of pressure movement helped reduce the static margin at rocket motor burnout, thereby increasing maneuverability at the coast condition without seriously impacting maneuverability during rocket motor burn. Simply moving the wings slightly forward might have had the same net effect at burnout, but this would have certainly changed the maneuverability more during rocket motor burn when initial acceleration commands are more rigorous (to get the interceptor on the correct course).

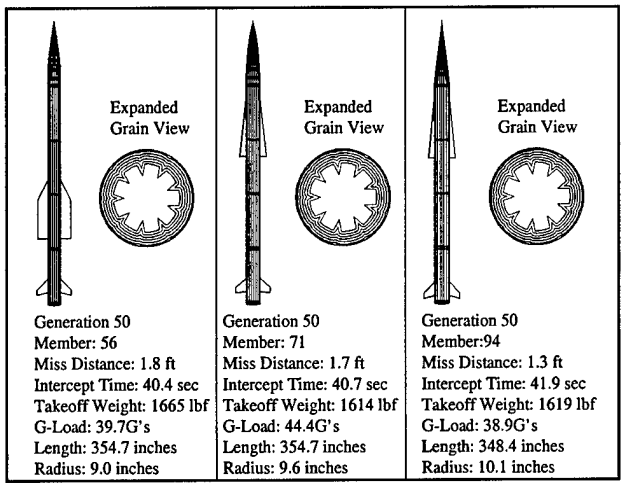

Figure 9. Three Generation 50 Designs with Miss Distances Less Than Two Feet

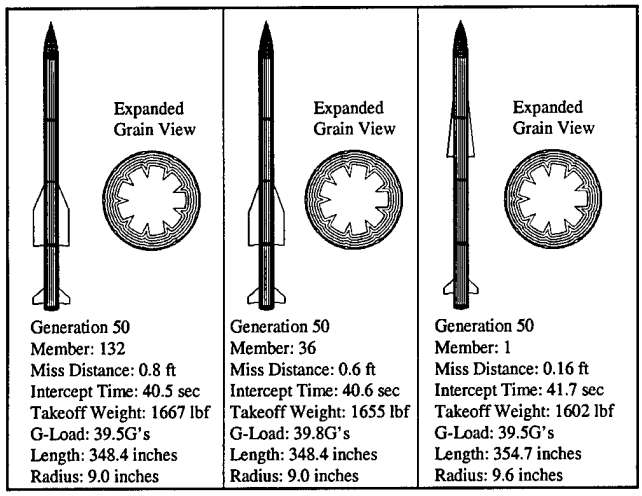

Figure 10. Three Generation 50 Designs with Miss Distances Less Than One Foot

To evaluate the performance of the types of interceptors designed by the genetic algorithm, the most accurate case (member #1 of generation 50) was selected for evaluation throughout the flight. Analysis of the interceptor trajectory showed a fairly straight path to the incoming target. It is interesting to note that right after launch the missile began a slight turn uprange from the initial launch angle. This turn is due to the fact that the missile has a very low velocity initially and the proportional navigation algorithm imparts an uprange command, thinking that the interceptor does not have enough velocity for a more direct path to the target. During the thrusting phase of the flight, the rocket motor imparts sufficient energy to the system to make the total temperature nearly reach the thermal limit, and the Mach number reaches approximately 4.1 at the 4.5 second burnout time. It is interesting that this burn-out time is nearly exactly that of at least one deployed missile system in the U.S. Arsenal.

It is obvious from Figure 11 why the interceptor is able to build up Mach 4 speeds so fast. The initial take-off thrust is nearly 64,000 lbf, and as the sharp star points burn off, the thrust reduces to approximately 37,000 lbf before increasing again as the burning area again begins to increase during the final burning phase.

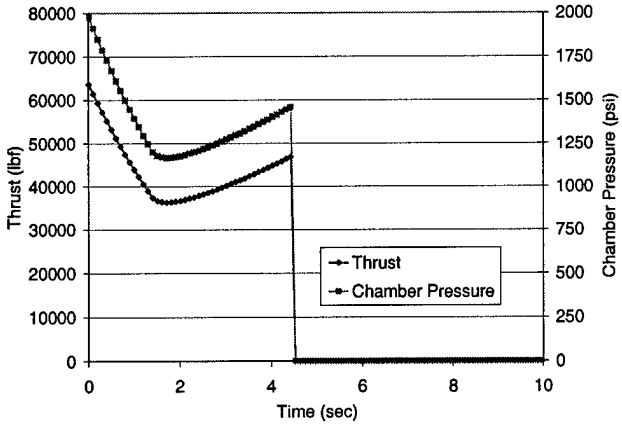

Figure 11. Thrust and Chamber Pressure History

One clear measure of thrust efficiency, however, is the ratio of the nozzle exit pressure to the local ambient pressure(P_e/P_a). A ratio of 1.0 is the most efficient, and Figure 12 shows that the genetic algorithm did a good job of compromising the exit pressure ratio to be both above and below the idea ratio during the burn. Since a variable nozzle exit area was not possible with this system, the ideal ratio cannot be maintained. It is remarkable, however, that the genetic algorithm was able to design the rocket motor throat/exit expansion ratio so that a reasonable pressure ratio was obtained while keeping the nozzle exit radius within the body case radius to avoid a drag penalty. The ability of the genetic algorithm to simultaneously work on multiple goals while under multiple constraints/penalties makes it a robust "hands-off" design tool. The fact that the genetic algorithm also seems to yield good results for implicit (not-stated) performance goals, such as maintaining a nozzle exit pressure ratio near 1.0, also highlights the value of the technique.

Computer Run Time

The 50 generations presented here required 10 days of CPU time of a Silicon Graphics R-10000 processor. The R-10000 is approximately 3 times faster than a 266MHz Pentium II processor. Given the amount of time spent on the computations and the amount of good design work the algorithm performed in that time, it is obvious why a learning algorithm is viewed as having a large potential for "all at once" design efforts.

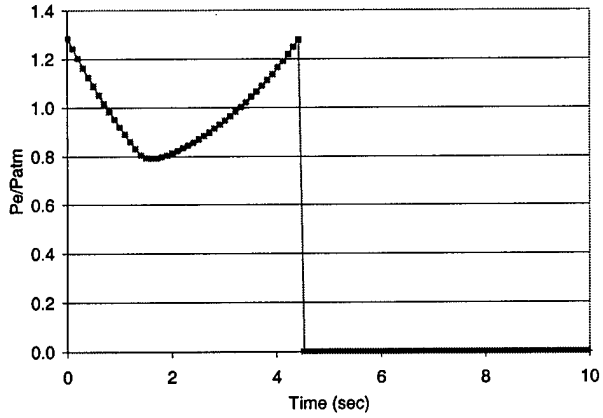

Figure 12. Exit Pressure Ratio History

Conclusions

The apportioned pareto genetic algorithm is well suited to designing complete interceptor systems consisting of propulsion, aerodynamics, and autopilot modules. This work has shown that an all-at-once design process, controlled by the proper artificial intelligence tool, is not only possible but much faster than iteratively designing each subsystem component into a workable package. Simple constraints, such as the thermal limits and structural integrity calculations used in this work, provide a means of injecting real-world considerations into the design process. Even with diverse performance modules and diverse goals, the genetic algorithm was able to learn how to design around the constraints while achieving good performance in overall system goals. In this difficult high-altitude/high-speed engagement scenario, the genetic algorithm developed multiple designs capable of close intercept.

The basic three-loop autopilot is ideal for preliminary design studies. Improvements such as thrust compensation could and should be included when the design process moves beyond the preliminary stage. The analytic determination of gains, and gain schedules based on Mach number and altitude, have made three-loop autopilots very popular in current missile systems. The basic proportional navigation guidance algorithm used in this work worked well, but it is likely that performance could be improved by using multiple algorithms during different phases of flight to correct, for example, the adverse acceleration commands at low launch speeds.

References

[1] Anderson, M.B., "Users Manual for IMPROVE© Version 2.2", 1999.

[2] Norris, S.R., and Crossley, W.A., "Pareto-Optimal Controller Gains Generated by a Genetic Algorithm," AIAA-98-1010, AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 1998.

[3] McGookin, E.W., Murray-Smith, D.J., Li, Y., and Fossen, T.I., "Parameter Optimisation of a Non-linear Tanker Control System Using Genetic Algorithms," Proceedings of the 2nd IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems, Glasgow, United Kingdom, 1997.

[4] Martin, R.C. IV, "A Gain Scheduling Optimization Method Using Genetic Algorithms," , Thesis, Air Force Institute of Technology, 1994.

[5] Karr, C.L., and Harper, T.R., "Genetic Algorithms in Adaptive Fuzzy Control," Conference Proceedings from The North American Fuzzy Logic Processing Society Meeting (NAFIPS 1992), Volume 2, pp. 506-514.

[6] Perhinschi, M.,G., "A Modified Genetic Algorithm for the Design of Autonomous Helicopter Control System," AIAA-97-3630, Presented at the AIAA Guidance, Navigation, and Control Conference, New Orleans, LA, August 1997.

[7] Karr, C.L., Freeman, L.M., and Meredith, D.L., "Genetic Algorithm based Fuzzy Control of Spacecraft Autonomous Rendezvous," NASA Marshall Space Flight Center, Fifth Conference on Artificial Intelligence for Space Applications, 1990.

[8] Homaifar, A., McCormack, E.D., "Full Design of Fuzzy Controllers Using Genetic Algorithms," NASA-N93-19452 06-80, The NASA Center for Aerospace Research at North Carolina A&T State University, 1992.

[9] Zarchan, Paul, *Tactical and Strategic Missile Guidance*, 3rd Edition, Volume 176, American Institute of Aeronautics and Astronautics, 1997.

[10] Nesline, F.W. and Nesline, M.L., "How Autopilot Requirements Constrain the Aerodynamic Design of Homing Missiles," Conference Volume of 1984 American Control Conference, San Diego, CA, June 6-8, 1984.

[11] Washington, W. D., "Missile Aerodynamic Design Program", 1980, 1990.

[12] Ashley, H., and Landahl, M., *Aerodynamics of Wings and Bodies*, Dover Publications Inc., New York, New York, 1965.

[13] Etkin, Bernard. *Dynamics of Atmospheric Flight*, John Wiley and Sons, 1972.

[14] Anderson, M.B., "Design of a Missile Interceptor Using Genetic Algorithms", Dissertation, Auburn University, Auburn, Alabama, 1998.